Journal of Information Hiding and Privacy Protection
DOI:10.32604/jihpp.2021.014108
Article

A Reversible Data Hiding Algorithm Based on Secret Sharing

Xin Jin*, Lanxin Su and Jitao Huang

School of Computer and Software, Nanjing University of Information Science & Technology, Nanjing, 210044, China
*Corresponding Author: Xin Jin. Email: talyorj@163.com
Received: 31 August 2020; Accepted: 16 December 2020

Abstract: In traditional secret sharing schemes, all shared images containing secret segments are needed to recover secret information. In this paper, a reversible data hiding scheme based on Shamir secret sharing is used. Secret information can be recovered even if only part of the encrypted sharing is owned. This method can reduce the vulnerability of traditional encryption sharing schemes to attack. Before uploading the secret information to the cloud server, embed the encrypted n segments of secret information into n different pictures. The receiver downloads t images from the cloud server (t < n), extracts the encrypted information using the watermark extraction algorithm, and obtains the original secret information after decryption through Shamir secret sharing.

Keywords: Feature gist; LSH; image retrieval

1 Introduction

In recent years, digital watermarking technology is widely used to protect secret information or judge whether the carrier has been tampered with, but many people ignore its function in the encrypted domain. Watermarking has served for decades as a very successful technique to secure the content in the plaintext domain but its potential in the encrypted domain has been explored to a lesser extent [1,2]. The following are examples of the application of digital watermarking in the literature referred to in this paper. In [3], a robust watermarking scheme implemented in the encryption domain is proposed. By combining the discrete wavelet transform with the discrete cosine transform, and using the partially homomorphic pallier cryptography system, they realized the security technology to resist attacks in the encryption domain. In [4], the encrypted image was partitioned into blocks, and blocks were extracted and recovered according to the descending order of absolute smoothness difference between the two candidate blocks, and the error rate was further reduced by using side matching technology. Shamir technology is gradually being used by scholars in conjunction with digital watermarking technology. In order to promote cloud-based multimedia systems, media information is securely distributed to multiple shamir-based and pob-based digital systems for secret sharing. Encrypting the processed images and sharing the domain itself eliminates the threat of any information leaking to a third-party server [5,6]. A secret image sharing scheme with additional steganographic and authentication functions was proposed based on Shamir method. In this scheme, the image is divided into n parts, hidden in the camouflage image, and the fragile watermark signal is embedded for authentication to form the password image. The scheme has an authentication function that can detect the Shared information of false participants before the end of the recovery process. Digital watermark is also used in different areas for the protection of the medical image, is proposed based on a stream cipher and the combination of block cipher encryption watermark system has introduced in [7], this scheme combined watermark/slower than simply encrypt the image encryption system, but it provides a reliable control function, image decryption execution are not affected. For the protection of commodity property rights, an interactive sale and purchase agreement for invisible watermarks is proposed in [8]. The seller does not know the watermarks received by the buyer, and the

buyer cannot obtain the original image information, thus protecting the privacy of both parties. The protocol can also be combined with different watermarking techniques and appropriate public key encryption techniques. In [9], the LSBs of the cover image are not directly replaced by secret data and authentication code. In order to improve the image quality, the optimal pixel adjustment method is adopted. The proposed method achieves these two purposes: high visual image quality and high authentication capability stego-image. Yuan proposed two kinds of secret sharing methods for natural overlay images, using multi-overlay adaptive steganography to adaptively share secret information between images [10]. The three papers of Zhang's team also gave many ideas on the digital watermark encryption scheme. At the beginning, Zhang proposed a reversible data hiding scheme of encrypted images with low computational complexity. With both the encryption key and the data hiding key, the decrypted image information can be obtained and the original image can be recovered [11,12]. The method has been improved to extract additional data and recover the original content by utilizing spatial correlation in natural images, except when the receiver has both encryption key and data hiding key [12]. In 2015, Zhang's team proposed a new reversible visible watermark encryption scheme, which USES pseudo-random data to encrypt the original image data bit by bit or by operation. Binary watermark images can be inserted and additional data can be embedded into the encrypted image by modifying the encrypted data [13].

2 Preliminaries

2.1 Shamir Threshold Secret Sharing

Secret sharing scheme is to divide secret information S into n secret information, store sub secret information and distribute it to n participants $\{P_1, P_2, \cdots, P_n\}$, among which,distribute S_i to participants P_i, only the subset of participants in the authorization set can use its own sub secret to recover.

In 1979, Shamir and Blakley proposed the first threshold secret sharing algorithm by using the methods of algebra and geometry. The scheme is as follows: given a positive integer k and n, $k \le n$,the (k,n) threshold secret sharing scheme is to divide the secret information S into n secret information, in which any k part of it or more sub secrets can reconstruct the secret information S, and any $k-1$part or less sub secrets cannot get any information of the secret information. Shamir's secret sharing scheme is widely used. The implementation process is as follows:

Select a finite field F_q, where $q \ge n$,set the set of participants as $P = \{P_1, P_2, \cdots, P_n\}$, k as threshold Secret information $S \in F_q$. Select n non-zero elements $x_1, x_2, \cdots, x_n$ from F_q which are different from each other and expose them.

Select $k-1$ polynomials from F_q randomly, $a_0 = S$, the rest a_i were randomly selected from F_q .Calculated $Si = g(x_i), i = 1,2,\cdots,n$ separately and sent (x_i, S_i) to members P_i as a sub secret.

Any member can share its sub secret and recover the secret information S by Lagrange interpolation. Set the sub secret of k members as $\{(x_{i_1}, S_{i_1}), \cdots, (x_{k_t}, S_{k_t})\}$, and the Lagrange interpolation formula is as follows:

$$g(x) = \sum_{r=1}^{k} S_{i_r} \prod_{\substack{t=1 \\ t \ne r}}^{k} \frac{x - x_{i_t}}{x_{i_r} - x_{i_t}} \tag{1}$$

According to the theory of polynomials, If the function values of two $k-1$ degree polynomials are equal at different values of variables, then the two polynomials must be equal, so formula (1) is established and calculated $S = a_0 = g(0)$.

2.2 Singular Value Decomposition (SVD)

Singular value decomposition (SVD) is a method of matrix decomposition, that is, a method of "disassembling a matrix into the product of several matrices". This method is often used in image processing because of its stability to general distortion and its ability to represent algebraic characteristics of images. Suppose M is an matrix of $m \times n$, in which all the elements belong to the domain K, the domain of real Numbers or complex Numbers. So there is a decomposition that makes $M = U\sum V^*$.

U is an unitary matrix of $m \times m$ order; $\sum$ diagonal matrix is positive semi-definite $m \times n$ order; And V^*, the conjugate transpose of V, is an unitary matrix of $n \times n$. Such a decomposition is called the singular value decomposition of M. $\sum$ diagonal elements on $\sum i$, where $\sum i$ is the singular values of M.

$$S = diag(\sigma_1, \sigma_2, \cdots, \sigma_r) \quad \sigma_1 \geq \sigma_2 \geq \sigma_3 \cdots \geq \sigma_r \tag{2}$$

σ_i is the singular value of the matrix whose rank is $r = \min(p, q)$.

2.3 The Overall Flow Chart

The following Fig. 1 is the functional flowchart of this software, which is divided into two parts: “encryption part” and “decryption part”. The sender and the receiver complete one of them, and they can achieve safe and reliable encrypted communication. First, the sender decomposes the ciphertext to be shared into n secret information through the shamir secret sharing method, and then embeds the n secret information into n images, and sends the n images. Here the “encryption part” is officially completed. Then the receiver only needs to receive k encrypted images, and then perform corresponding extraction operations through the software to recover the ciphertext to obtain complete ciphertext information. Here the “decryption part” is completed. The following is a functional flowchart of this software, which is divided into two parts, namely “encryption part” and “decryption part”. First, the sender decomposes the ciphertext to be shared into n secret information through the shamir secret sharing method, and then embeds the n secret information into n images, and sends the n images. Here the “encryption part” is officially completed. Then the receiver only needs to receive k encrypted images, and then perform corresponding extraction operations through the software to recover the ciphertext to obtain complete ciphertext information. Here the “decryption part” is completed.

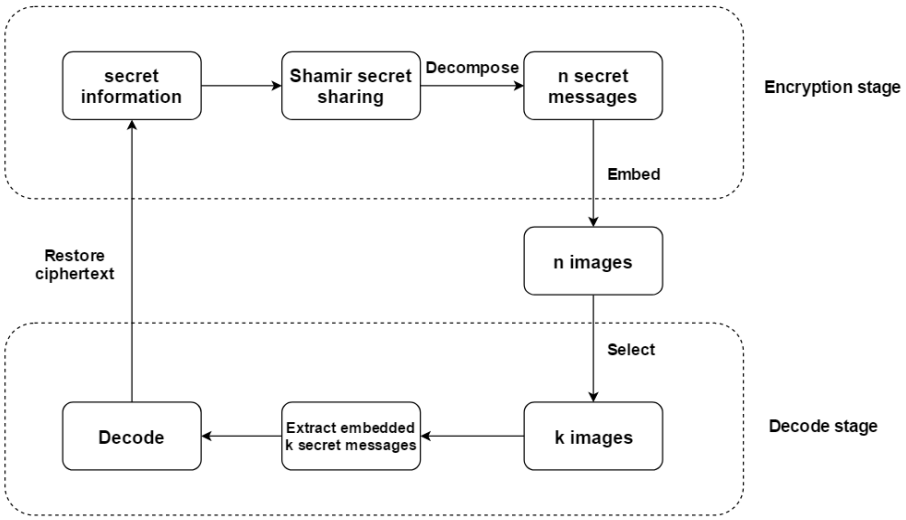


Figure 1: The overall flow chart

3 Expressions Experimental Result and Discussion

3.1 Solution Process

In-line equations/expressions are embedded into the paragraphs of the text. For example, $E = mc^2$. In-line equations or expressions should not be numbered and should use the same/similar font and size as the main text.

The scheme proposed in this paper provides a secure method for transmitting secret information through images. First given A secret information, set the threshold k and n is the number of secret information, and generate A random key A, the method of using shamir generate n different sub secret information, take advantage of cloud architecture of high-end, before upload photos to the cloud server, set the given A key B by means of the JPEG based on DCT domain jsteg steganographic algorithm embedding secret information is encrypted. The media information that needs to be transmitted is then spread over multiple encrypted Shared images that are visually identical to the original image. Only users with specific key B, random key A, and at least k subsecret information can extract secret information from these encrypted images.

Figs. 2 and 3 are the decomposition ciphertext and the synthesis ciphertext respectively. Fig. 2 input A secret information to be transmitted, and then you can get the key A generated by the system. At the required threshold value k and the required number of subkeys n, the program based on shamir will output n subcipher segments (no repetition, repetition will be generated again). The subkey extraction section is complete. Fig. 3 shows the selected k different encrypted images, and then extract their sub-cipher segments respectively. Input k sub-cipher segments successively into the shamir-based system, and the original complete password information can be generated.

```
select 1 for disassembling secrets, choose 2 for restoring secrets, and choose 3 for exit
1
Input your secret
20178314
modulus：39979999
Input threshold
3
Input the number of subkeys
5
The subkey is (make sure the subkey is not duplicated, please regenerate if any)
22 1899690
87 16254466
86 16294093
59 3561820
51 34657367
select 1 for disassembling secrets, choose 2 for restoring secrets, and choose 3 for exit
```

Figure 2: Shamir encrypt

```
select 1 for disassembling secrets, choose 2 for restoring secrets, and choose 3 for exit
2
Input modulus
39979999
Input threshold
3
 Please input the key
(first input all points, then enter the corresponding value of each point)
22 87 86 1899690 16254466 16294093
The secret information is 20178314
select 1 for disassembling secrets, choose 2 for restoring secrets, and choose 3 for exit
```

Figure 3: Shamir decode

3.2 Opening Animation

First of all, when we click to run this software, a cut-in animation will pop up on the GUI interface, the animation process such as Figs. 4 and 5. This animation is to draw a motion trajectory using the matlab GUI, and finally simulates a celestial body running image that the sun, moon, and satellites move around the corresponding axis.

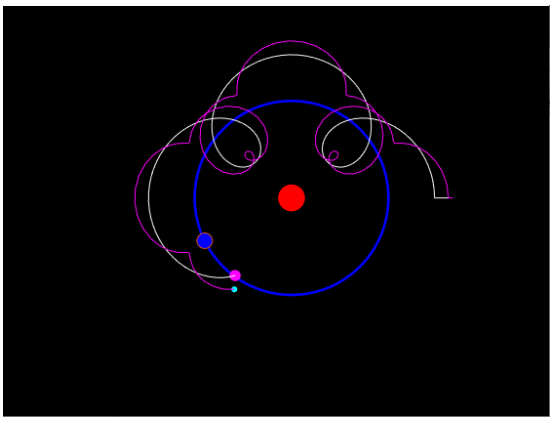


Figure 4: Opening animation in progress

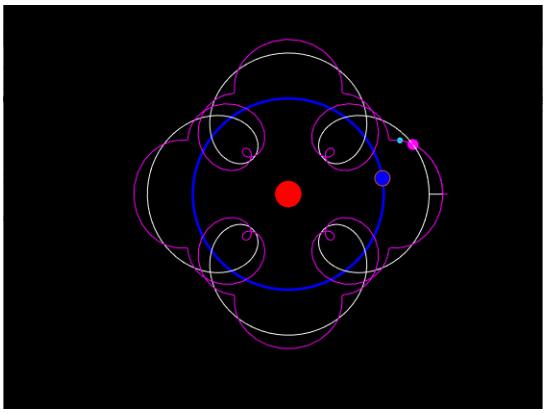


Figure 5: Opening animation is finished

3.3 Main Menu

After playing the cutscenes, the software will jump to the main menu interface, such as Fig. 6. The main menu interface is also designed through the matlab GUI. When you come to the main menu, you can see four buttons, namely “image encryption”, “filter”, “add noise”, “exit”. Except that the button “exit” is used to close the software interface, the other three are the core functions of the software, which will be described in detail one by one.

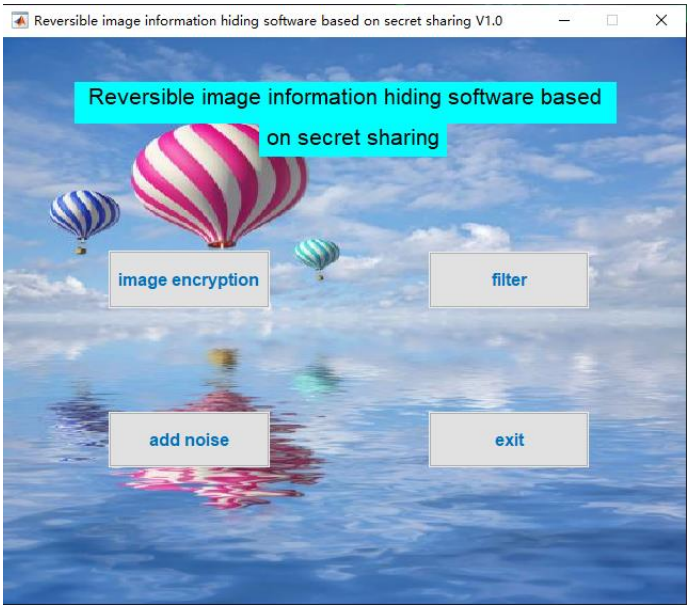


Figure 6: Main menu interface

3.4 Image Encrypting

Click the “image encryption” button to enter the function interface, such as Fig. 7. Then you can click “original image” to select the image to be encrypted. The corresponding image will be displayed in the “Cover” box in the upper left corner of the GUI interface. Then click “hidden text” to select the text file that needs to be embedded. This text contains one of the five ciphertext segments that were encrypted by shamir method. After selecting, the specific information will be in the lower left the corner of the “Message’ box is displayed. Set a key value yourself in the “key” box, used to set the key for embedding and extracting information. Then click “embed Information”, after the software pop-up window prompts “embed successfully!”, it means the embedding is completed, such as Fig. 8. If you

need to extract the hidden information in the encrypted image, you need to click “hidden image” to select the encrypted image, and then the image will be displayed in the “Stego” box in the upper right corner. Enter the correct key value in the “Key” box and click “extract information”. The previously embedded information is displayed in the “Message” box in the lower right corner, such as Fig. 9. Then there are two function buttons “restart” and “exit”. The button “restart” will reset this interface, and will delete the encrypted image and the extracted secret information, which also ensures the security of the information. “exit” will directly close the software.

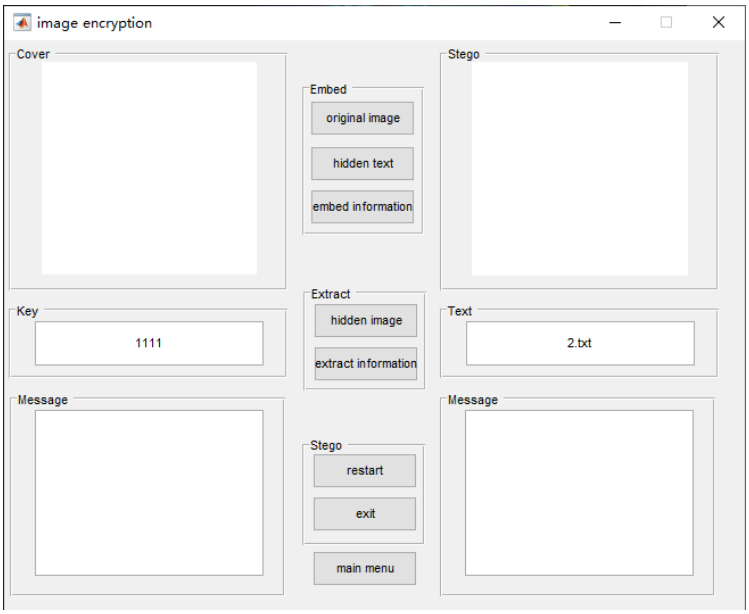


Figure 7: The initial encryption function interface

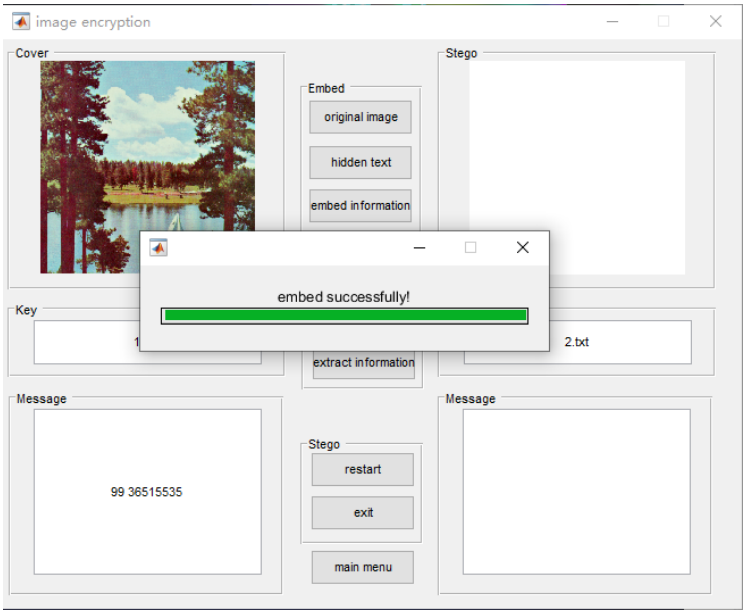


Figure 8: Information has been successfully embedded

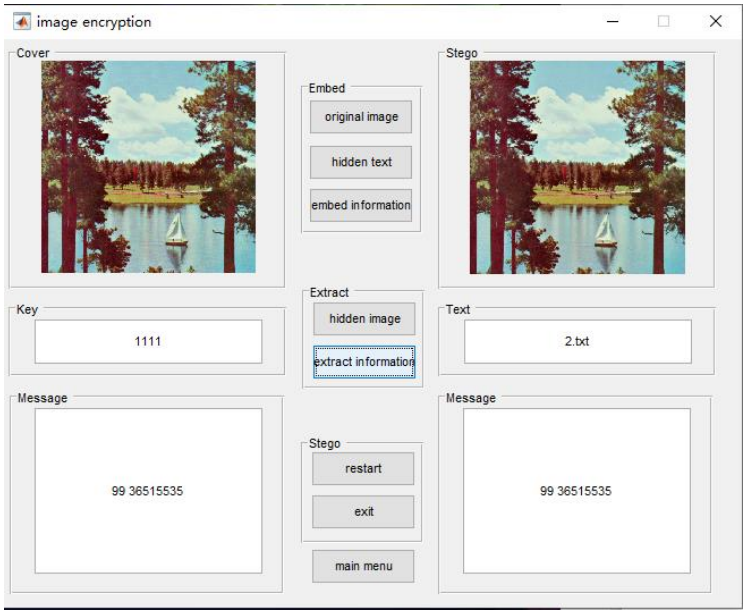


Figure 9: Information has been successfully extracted

3.5 Filtering

First, on the main menu interface, click the "filter" button. Then it will enter the "filter" interface, where you can perform "Gaussian filter", "Laplace filter", and "Wiener filter" on any image, such as Figs. 10–12, and the comparison between the original image and the processed image will be displayed in the interface. This function is used to verify that the image after embedding the information is filtered, and then extract the encrypted information to see whether the result will affect the information extraction. After conducting experiments, it is found that the encrypted image after filtering can still extract the correct secret information. It can be seen that the method of embedding and extracting information in this paper has a certain ability to resist the filtering of the encrypted image.

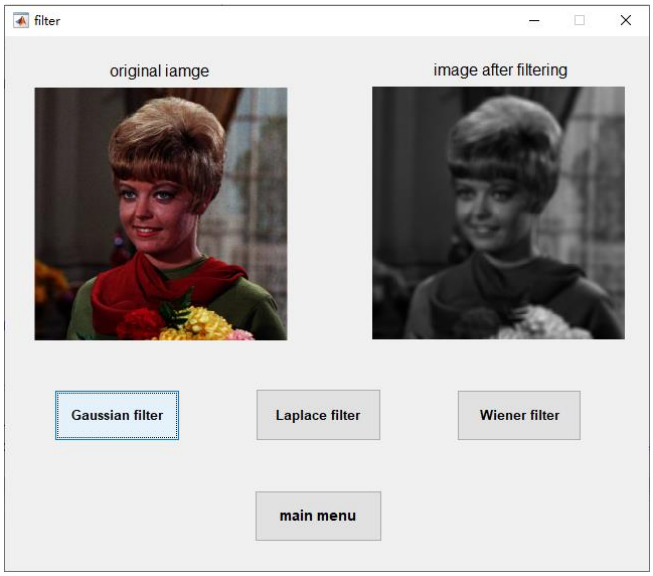


Figure 10: Comparison before and after Gaussian filtering

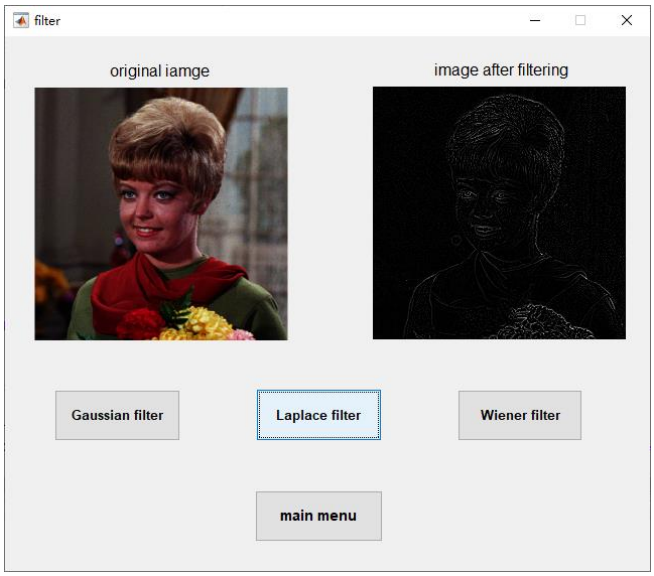


Figure 11: Comparison between before and after Laplace filtering

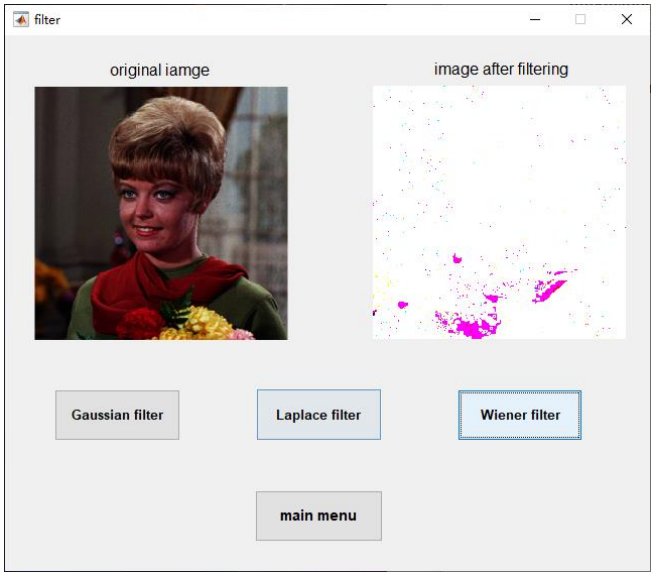


Figure 12: Comparison before and after Wiener filtering

3.6 Noise Processing

First, on the interface of the main menu, click the “add noise” button. Then it will enter the “Multiplicative noise” interface, where you can perform “Multiplicative noise”, “Salt and pepper noise”, and “Gaussian noise” on any image, such as Figs. 13–15, and the comparison between the original image and the processed image will be displayed in the interface. This function is used to verify that the image after embedding the information is processed for noise, and then extract the encrypted information to see whether the result will affect the information extraction. After conducting experiments, it is found that the encrypted image after the noise attack can still extract the correct secret information. It can be seen that the method of embedding and extracting information in this paper has a certain ability to resist the encryption image from being attacked by noise.

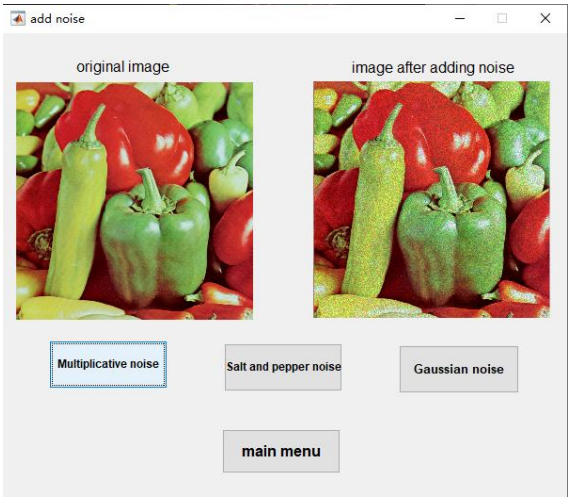


Figure 13: Multiplicative noise before and after comparison

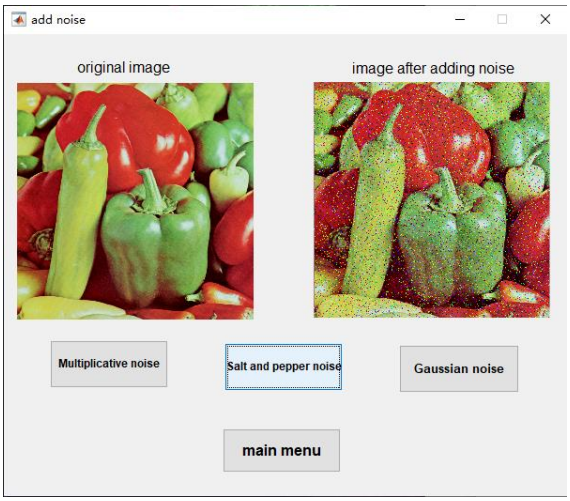


Figure 14: Salt and pepper noise before and after comparison

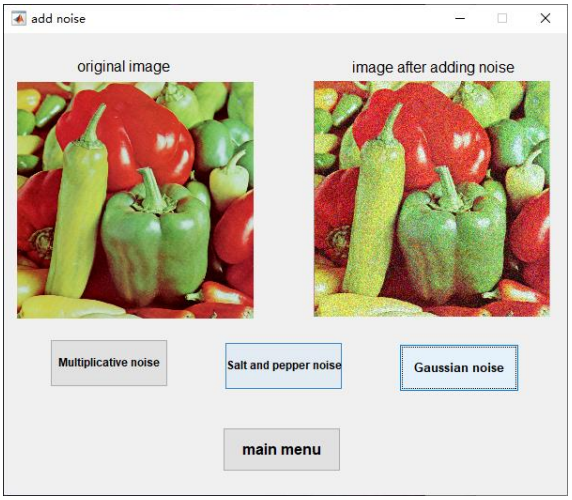


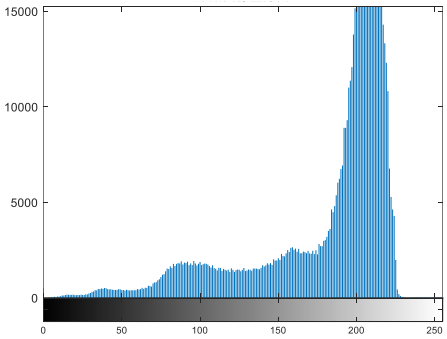
Figure 15: Gaussian noise processing before and after comparison

3.7 Image Testing

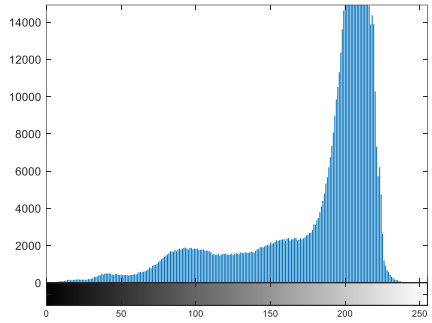
In Figs. 16(a)–16(h), we selected two original images and their encrypted images for comparison, as well as their histogram comparison. It can also be seen from the histogram, because the two differences are very small, it is impossible to judge whether they have ciphertext embedding.

(a) Original image

(b) Image after embedding information



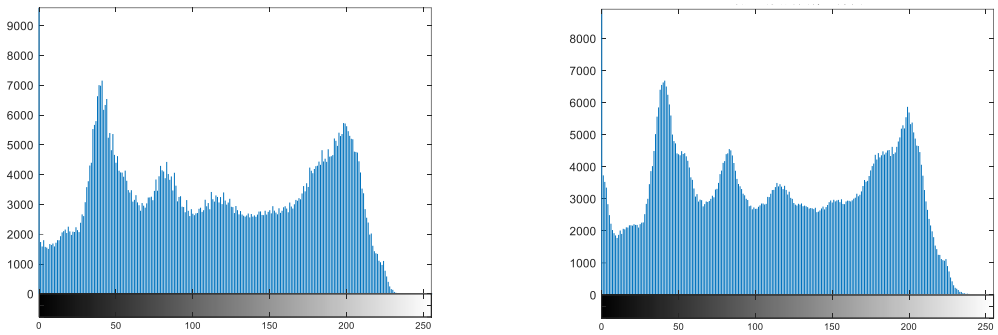
(c) Histogram of original image



(d) Image after embedding information

(e) Original image

(f) Image after embedding information



(g) Histogram of original image (h) Histogram after embedding information

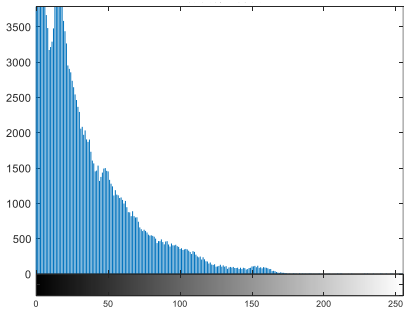
Figure 16: Histogram before and after embedding information

Although the proposed encryption scheme makes the contents unreadable in the encryption domain, thus protecting the sub-codices embedded in the image, the data is still vulnerable to attacks by remote distributed cloud data centers. An attacker may attempt to maliciously tamper with the Shared image without knowing the actual content. In order to visualize the impact of the attack on the embedded secret information and the recovery of the original content, the following situations are simulated to detect the attack on the image. In Fig. 17, an encrypted image (a) is selected to be attacked by pepper and salt noise with noise density of 0.09 (b), multiplicative noise with noise density of 0.02 (c) and gaussian noise with noise density of 0.06 (d), (e) to (h) are histogram comparisons of images before and after adding noise. It can be intuitively seen that the histogram of the encrypted Shared image is similar to the histogram of the original image, so the third party cannot extract the information of the original image from the Shared image, reflecting the security of this method.

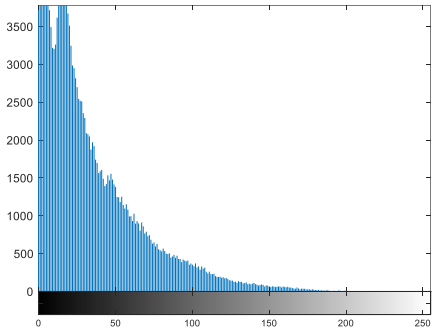
(a) Original image (b) Add multiplicative noise with noise density of 0.02

(c) Add pepper and salt noise with noise density of 0.09

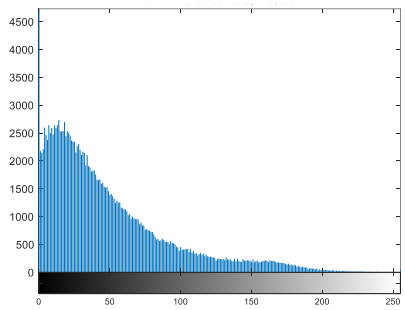
(d) Add gaussian noise with noise density of 0.06



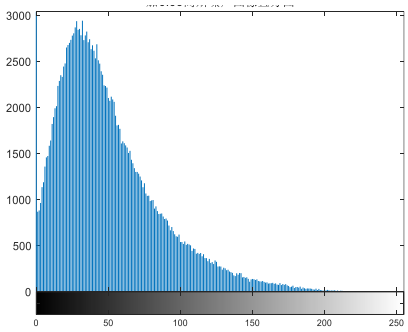
(e)Histogram of original image



(f) Histogram after adding multiplicative noise



(g) Histogram after adding salt and pepper noise



(h) Histogram after adding Gaussian noise

Figure 17: Histogram before and after adding noise

In Fig. 18, different filters are used to process the encrypted image, and then the ciphertext segment is extracted. Finally, it is found that median filtering, gaussian filtering and mean filtering are used for the encrypted image. You end up with the right ciphertext.

(a) Gaussian filter (b) Laplace filter (c) Wiener filter

Figure 18: Different filters are used to process the image

Table 1: Five encrypted images were selected and compared with the original images based on PSNR

Image	PSNR
Image1	36.8210
Image2	37.4755
Image3	35.2209
Image4	38.8758
Image5	36.9491

4 Conclusions

In order to reduce the vulnerability of multimedia content residing on remote cloud servers, a reversible data encryption domain hiding scheme based on secure sharing is proposed. In this scheme, shamir encryption algorithm is used to divide ciphertext into n subsecret, and jsteg steganographic algorithm in DCT domain is used to embed n subsecret information into multiple image sharing, thus making media information fuzzy. Only the legitimate owner of the key can obtain t or more images to extract the information after the media recovery. The robustness of the secret information in the encrypted domain is tested for different attack scenarios. The encrypted image can withstand median filter, gaussian filter, mean filter, and some degree of gaussian noise, pepper and salt noise and multiplicity noise. The results show that the scheme is robust. The scheme is fault-tolerant enough to handle some cloud server outages because confidential owner information can be extracted from other servers. Unless more than n-k cloud centers are attacked, secret recovery will not be affected and the correct secret information can still be extracted.

Funding Statement: There is no fund support for this paper.

Conflicts of Interest: We declare that we do not have any commercial or associative interest that represents a conflict of interest in connection with the work submitted.

References

[1] K. Shishir, N. Jain and S. L. Fernandes, "Rough set based effective technique of image watermarking," *Journal of Computational Science*, vol. 19, pp. 121–137, 2016.

[2] C. Qin, H. Wang, X. Zhang and X. Sun, "Self-embedding fragile watermarking based on reference-data interleaving and adaptive selection of embedding mode," *Information Sciences*, vol. 373, pp. 233–250, 2016.

[3] J. Guo, P. Zheng and J. Huang, "Secure watermarking scheme against watermark attacks in the encrypted domain," *Journal of Visual Communication and Image Representation*, vol. 30, pp. 125–135, 2015.

[4] W. Hong, T. Chen and H. Wu, "An improved reversible data hiding in encrypted images using side match," in *IEEE Signal Processing Letters*, vol. 19, no. 4, pp. 199–202, 2012.

[5] P. Singh, B. Raman and M. Misra, "Just process me, without knowing me: A secure encrypted domain processing based on Shamir secret sharing and POB number system," *Multimedia Tools and Applications*, vol. 77, no. 2, pp. 1–25, 2017.

[6] C. C. Lin and W. H. Tsai, "Secret image sharing with steganography and authentication," *Journal of Systems and Software*, vol. 73, no. 3, pp. 405–414, 2004. DOI 10.1016/S0164-1212(03) 00239-5.

[7] D. Bouslimi, G. Coatrieux, M. Cozic and C. Roux, "A joint encryption/watermarking system for verifying the reliability of medical images," *IEEE Transactions on Information Technology in Biomedicine*, vol. 16, no. 5, pp. 891–899, 2012. DOI 10.1109/TITB.2012.2207730.

[8] N. Memon and P. W. Wong, "A buyer-seller watermarking protocol," in *1998 IEEE Second Workshop on Multimedia Signal Proc.*, 1998, pp. 291–296. DOI 10.1109/MMSP.1998.738949.

[9] C. C. Wu, S. J. Kao and M. S. Hwang, "A high quality image sharing with steganography and adaptive authentication scheme," *Journal of Systems and Software*, vol. 84, no. 12, pp. 2196–2207, 2011.

[10] H. D. Yuan, "Secret sharing with multi-cover adaptive steganography," *Information Sciences*: *An International Journal*, vol. 254, pp. 197–212, 2014. DOI 10.1016/j.ins.2013.08.012.

[11] X. Zhang, "Reversible data hiding in encrypted image," *IEEE Signal Processing Letters*, vol. 18, no. 4, pp. 255–258, 2011. DOI 10.1109/LSP.2011.2114651.

[12] X. Zhang, "Separable reversible data hiding in encrypted image," *IEEE Transactions on Information Forensics and Security*, vol. 7, no. 2, pp. 826–832, 2012.

[13] X. Zhang, Z. Wang, J. Yu and Z. Qian, "Reversible visible watermark embedded in encrypted domain," *2015 IEEE China Summit and International Conference on Signal and Information Processing*, 2015, pp. 826–830. DOI 10.1109/ChinaSIP.2015.7230520.